Semantic Analysis of Web Site Audience by Integrating Web Usage Mining and Web Content Mining

Jean-Pierre Norguet[1], Esteban Zimányi[1], and Ralf Steinberger[2]

[1] Université Libre de Bruxelles
Laboratory of Computer and Network Engineering, CP165/15
Avenue F.D. Roosevelt, 50
1050 Brussels, Belgium
[2] European Commission – Joint Research Centre
Via E. Fermi 1, T.P. 267
21020 Ispra (VA), Italy
http://www.jrc.it/langtech

Abstract. With the emergence of the World Wide Web, analyzing and improving Web communication has become essential to adapt the Web content to the visitors' expectations. Web communication analysis is traditionally performed by Web analytics software, which produce long lists of page-based audience metrics. These results suffer from page synonymy, page polysemy, page temporality, and page volatility. In addition, the metrics contain little semantics and are too detailed to be exploited by organization managers and chief editors, who need summarized and conceptual information to take high-level decisions. To obtain such metrics, we propose a method based on output page mining. Output page mining is a new kind of Web usage mining, between Web usage mining and Web content mining. In our method, we first collect the Web pages output by the Web server. Then, for a given taxonomy covering the Web site knwoledge domain, we aggregate the term weights in the output pages using OLAP tools, in order to obtain topic-based metrics representing the audience of the Web site topics. To demonstrate how our approach solves the cited problems, we compute topic-based metrics with SQL Server OLAP Analysis Service and our prototype WASA for real Web sites. Finally, we compare our results against those obtained with Google Analytics, a popular Web analytics tool.

Keywords: World Wide Web, Web analytics, Semantic Web, Web usage mining, Data Mining.

1 Motivations and Related Work

With the emergence of the Internet and of the World Wide Web, the Web site has become a key communication channel in organizations. To satisfy the objectives of the Web site and of its target audience, adapting the Web site content to the users' expectations has become a major concern. In this context, Web usage mining, a relatively new research area, and Web analytics, a part of Web

I.-H. Ting, H.-J. Wu (Eds.): Web Mining Appl. in E-Commerce & E-Services, SCI 172, pp. 65–79.
springerlink.com

usage mining that has most emerged in the corporate world, offer many Web communication analysis techniques. These techniques include prediction of the user's behaviour within the site, comparison between expected and actual Web site usage, adjustment of the Web site with respect to the users' interests, and mining and analyzing Web usage data to discover interesting metrics and usage patterns [1]. However, Web usage mining and Web analytics suffer from significant drawbacks when it comes to support the decision making at the higher levels in the organization.

Indeed, according to organizations theory [2], the higher levels in the organizations need summarized and conceptual information to take fast, high-level, and effective decisions. For Web sites, these levels include the organization management and the Web site chief editor. At these levels, the results produced by Web analytics tools are mostly useless. Indeed, most reports target Web designers and Web developers [3]. Summary reports like the number of visitors and the number of page views can be of some interest to the organization manager but these results are poor. Finally, page-group hits give the Web site chief editor conceptual results, but these are limited by several problems like page synonymy (several pages contain the same concept), page polysemy (a page contains several concepts), page temporality, and page volatility. These limitations therefore make Web analytics tools mostly useless to this problem domain.

Web usage mining research projects have mostly left Web analytics aside and have focused on other research paths like usage pattern analysis, personalization, system improvement, site structure modification, marketing business intelligence, and usage characterization [1]. Usage pattern analysis aims to discover interesting usage patterns to understand the needs of the Web site and better serve visitor satisfaction [4]. Personalization provides dynamic recommendations to visitors based on their profile [5]. System improvement uses Web usage mining to develop policies for Web caching, network transmission, or load balancing in order to optimize Web site performance and quality of service [6]. Site structure modification provides feedback on visitor behaviour to reorganize content among pages and optimize hyperlinks [7]. Business intelligence efforts integrate Web usage data with marketing data and use OLAP query tools to improve customer attraction, customer retention, and cross sales [8]. Finally, usage characterization monitor client-side activity to understand and predict Web site browsing strategies [9]. All these domains have proven very fertile but have provided little contributions to Web analytics.

An interesting contribution was attempted with reverse clustering analysis [10], a technique based on self-organizing feature maps. This technique integrates Web usage mining and Web content mining to rank the Web site pages according to an original popularity score. However, the algorithm is not scalable and does not answer the page-polysemy, page-synonymy, page-temporality, and page-volatility problems. An interesting attempt to solve these problems is proposed in the IUNIS algorithm of the Information Scent model [11]. This algorithm produces a list of term vectors representing the users' needs, which can be easily interpreted. On the other hand, the results are visit-centric rather than

site-centric, suffer from term polysemy and term synonymy, and the algorithm scalability is unclear. Finally, according to a recent survey [12], no Web usage mining research project has proposed a satisfying solution to provide site-wide summarized and conceptual audience metrics.

To answer the need of such metrics, our approach aims at analyzing the Web content output by Web servers. Indeed, so far, little or no interest has been shown in the content of the output pages. This disinterest is explained by the lack of techniques to collect the output Web pages and by the high number of pages to analyze afterwards [1]. We therefore provide the necessary means to collect the output pages and then to analyze the mined content. In Section 2, we present the methods that we conceived to collect the output pages: content journaling, script parsing, server monitoring, network monitoring, and client-side collection. These methods should allow to collect the output pages of any Web site. In Section 3, we explain how term weights in these pages can be aggregated with respect to a taxonomy representing the Web site domain knowledge domain in order to obtain audience metrics representing the consultation, presence, and visitors' interest into the Web site topics. In Section 4, we present and discuss the results obtained with SQL Server OLAP Analysis Service and our prototype WASA on several case studies. In particular, we compare different metrics, we show some interesting visualizations, we study the effect of the taxonomy knowledge domain, and we validate our approach against Google Analytics, a popular Web analytics tool. Finally in Section 5, we describe the results exploitation process, we expose the limitations of the approach, and we present some insights of solutions for future work.

2 Output Page Mining

The first step in our approach is to mine the Web pages that are output by the Web server. To this end, we have conceived a number of methods, each of them being located at some point in the Web environment (Figure 1). The Web environment is centered around the Web server hosting the Web site. The Web server is connected to the Internet Service Provider network, which is connected to the Internet via a router. At the other extremity of the Internet, visitors connect the Web site using their browser.

The output Web pages can be collected at several points in the Web environment: (1) the Web server file system, (2) the Web server running instance, (3) the network wire, and (4) the client-side machine. We call the corresponding collection methods (1) Web logs and content journaling, (2) server monitoring, (3) network monitoring, and (4) client-side collection. These collection points are similar to the meta-data collection points used in Web analytics tools. The main difference is the complexity of the collection methods; accessing the page content requires more efforts than regarding only the communication meta-data. In the next sections, we describe and discuss the collection method for each of the Web environment points.

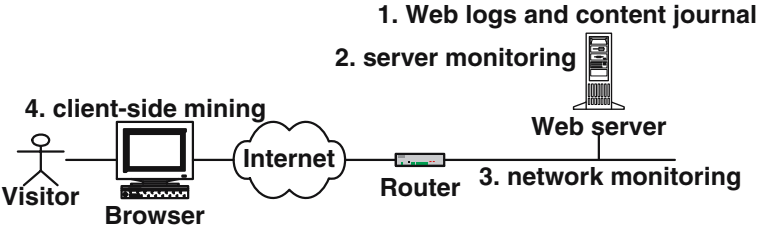

Fig. 1. Collection points in Web environment

2.1 Log Files and Content Journaling

The Web server file system is the most used and simplest collection point in Web analytics tools. Web server log files contain the references of each output Web page, so it is possible to retrieve the page content by looking up the associated file in the Web server document directory. However, if the page content evolves over time, the page version at analysis time can be different of the page version at consultation time.

Most log formats store the request time along with the page reference. So, it is possible to retrieve at analysis time the content of a page as it has been output at consultation time, from the request time, from the page reference, and from a journal that stores the temporal evolution of the pages. To keep track of temporal evolution of the pages, we schedule a daily batch that maintains a *content journal*. Practically, the content journal is a list of entries that are made of a URI, a time period, and a reference to the archived file. This allows to retrieve at any time the exact content of a viewed page, even if the content of the online page has changed over time.

This method requires to store the least amount of pages and subsequently requires the least amount of computation to obtain the various metrics presented in Section 3. As dynamic pages are unique and volatile, content journaling works for static Web pages only.

2.2 Script Parsing

As seen in the previous section, a content journal can only be produced for static pages. In many Web servers, dynamic pages are generated from scripted pages, which usually hardcode a part of the text content while the rest is retrieved from a database. We could therefore write a compiler that takes the scripted pages as input and removes the script instructions to produce a pure-HTML page with the hardcoded content.

For experimentation, we implemented [13] a script-parsing compiler for Java Server Pages. The result has proven satisfying as long as the scripted pages hardcode most of the content. If more content is externalized, the extracted content is reduced. Also, the compiler depends on the page scripting language and on the scripting language version. In the conclusion of our study, script parsing was abandoned in favor of server monitoring (Section 2.3).

2.3 Server Monitoring

Server monitoring collects the output pages within the Web server instance. Major Web servers offer an API to interact with the Web server kernel. This allows to insert a custom plugin that saves the output pages onto the file system or into a local or remote database. Practically, a server-monitoring plugin registers with the Web server kernel and gets the control on the output data after it has been sent to the browser. With this method, any kind of file can be traced, including dynamic pages.

Plugins are executed within the Web server. This introduces crash risk into the Web server. Such a risk may be an obstacle to its adoption in critical Web servers. In addition, server monitor plugins depend on the Web server API. Different Web servers therefore require different ports of the plugin. However, the porting efforts are reduced by the fact that two products dominate the Web server market: Apache HTTPD and Microsoft IIS.

On the other hand, Web server plugins can access HTTP headers, which include the request file extension and the MIME type of output files. This allows to store dynamic Web pages only, ignoring binary files and static Web pages. Another advantage is that server monitors can transform the response before it is sent to the browser. For example, combination of output page tracing with data compression has proven to save bandwidth and reduce response time [13].

We tested server monitoring by developing *mod_trace_output*, a server-monitoring plugin for the Apache Web server.[1] The plugin traced output Web pages with success and passed robustness, performance, and scalability tests. More details about the plugin architecture, implementation, and benchmarks can be found in [13].

2.4 Network Monitoring

A network monitor runs in a network-promiscuity mode on the same Ethernet network as the Web server and captures the TCP/IP packets on the network wire. To reassemble Web pages from the TCP/IP packets, a network monitor realizes the following action steps: (1) store TCP/IP packets, (2) filter and group the packets of each HTTP transaction, (3) sort and concatenate the packets to rebuild the transaction, (4) get the metadata from the HTTP header, and (5) remove the header from the HTTP response header to get the Web page.

A network monitor introduces no risk in the Web server. In addition, it is independent from the Web server brand or version. On the other hand, network monitoring is CPU-intensive because it needs to sort and concatenate many characters strings. In addition, all files are reassembled before the file type can be known, therefore CPU time is spent to capture unrelevant files like images. Network monitoring works for those networks that send the packets on the wired line, like Ethernet networks, and the network monitor must be on the same

[1] mod_trace_output is available as a SourceForge project: http://trace-output.sourceforge.net/

subnet as the Web server. Finally, network monitoring cannot read encrypted conversations from secure Web servers.

2.5 Client-Side Collection

In client-side collection, a program is embedded in the output Web page and runs inside the visitors' browser. When the page is loaded, the program runs inside the browser; it parses the page and sends the page content to a dedicated server, which stores the pages content (Figure 2).

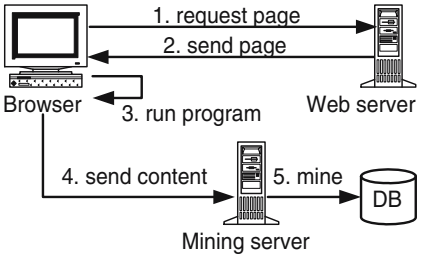

Fig. 2. Client-side collection

As the workload is distributed among the visitors' machines, this collection method can support high-traffic Web sites. To benefit of additional workload distribution, the embedded program can implement content processing (see Section 3). Client-side collection must be used when the publishing technologies involve page layout transformation in the browser, like client-side XML/XSL pages.

Visibility of client-side collection can be a problem. Indeed, visitors can feel unhappy to see that a program is running on their computer, is monitoring their pages, and is sending information to an unknown server. Another drawback of the method is the lack of control on the client side: evil visitors can tweak the program locally and send fake data to the collection server.

2.6 Summary

Most, if not all of the Web sites, can be handled by the above collection methods. Log file parsing combined with content journaling is a method that is easy to setup, runs in batch, and offers good performance. For dynamic Web sites, and when script parsing is not satisfying, the alternatives are server monitoring, network monitoring, and client-side collection. Server monitors are usually installed in secure Web sites, and network monitors elsewhere because of the lower risk. Client-side XML/XSL Web pages must be collected from the client browser. The pros and cons described in each of the previous sections should help choose a method or combination of methods for any Web site, whatever the Web-server or content-publishing technologies.

3 Topic-Based Audience Metrics

For the given Web site to analyze, we choose a taxonomy that models the Web site knowledge domain. The top terms in the taxonomy should represent the Web site the main Web site topics. Thus, for each taxonomy term, the term weight [14] in the output pages gives an indication of the term consultation by the visitors during the mining period. If the Web site is mostly static, the term weight in the online pages gives an indication of the term presence on the site. Term consultation and term presence are two interesting metrics but suffer from term polysemy and term synonymy. Also, the terms are too numerous to providde summarized results.

These problems can be overcome by aggregating the term metrics along the taxonomy. Indeed in most taxonomies, the terms are hierarchically linked by a relationship of type *is a* or *part of* [15]. In these taxonomies, the audience of the subterms contributes to the communication of the topics denoted by the superterms. Therefore, the audience metrics aggregation from the leaves up to the taxonomy root gives an indication of the audience obtained by the Web site topics. Furthermore, the consultation-to-presence ratio gives an indication of the visitors' *interest* into the topics. If the top terms in the taxonomy represent the Web site main topics, the corresponding consultation, presence and interest metrics can be used as conceptual audience measures.

For example, an e-commerce Web site selling food products might use an ontology where the food topic is divided into vegetable and fruit, which are in turn divided into potato and carot, and into apple and strawberry (Figure 3). The consultation and presence metrics for every terms are represented under the ontology nodes. Metrics aggregation from the leaves up to the root provides topic-based metrics. For example, the aggregated consultation for the *fruit* topic is given by the addition of the consultation for the terms {fruit, apple, strawberry}. The same is done for every topic, as well as for the presence metrics. The interest values are obtained by dividing the consultation and presence values for each topic.

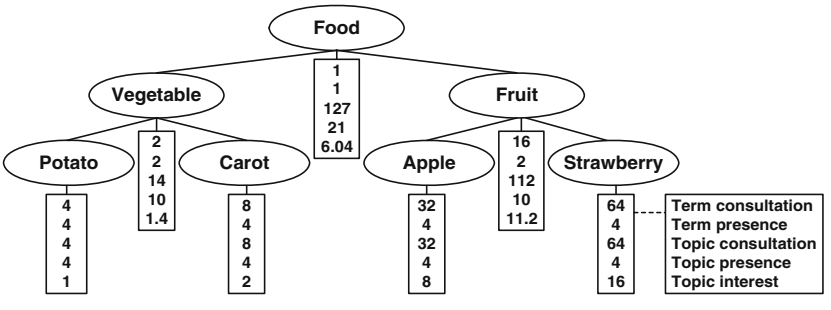

Fig. 3. Hierarchical aggregation

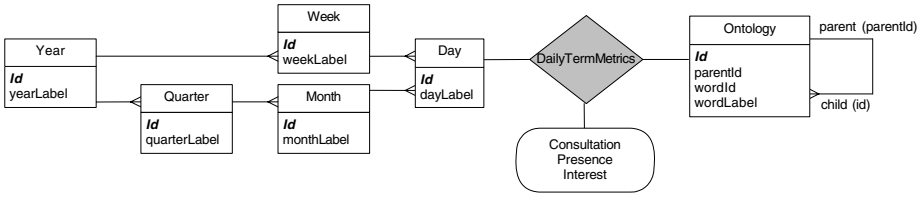

Fig. 4. OLAP cube with two dimensions: Time and Ontology

For a given mining period comprised between days d_1 and d_2 and a given topic T_i defined as the union of the term s_i and of its subterms in the taxonomy, we can formalize the topic consultation and presence metrics as follows:

$$\text{Consultation}(T_i, d_1, d_2) = \sum_{s_j \in T_i} \sum_{d=d_1}^{d_2} w_j(d) \tag{1}$$

$$\text{Presence}(T_i, d_1, d_2) = \sum_{s_j \in T_i} \int_{d_1}^{d_2} w'_j(t)dt \tag{2}$$

where $w_j(d)$ is the term weight of term s_j in the output pages mined during day d and $w'_j(t)$ is the term weight of term s_i in the online pages at time t. If the pages p_k have been online during a time Δt_k between d_1 and d_2, the integral is equal to $\sum_{p_k} w'_{jk} \Delta t_k$, where w'_{jk} is the weight of term s_j in page p_k. This expression can be computed easily.

Recursive aggregation of the term-based metrics into topic-based metrics can be computed by OLAP tools. This computation requires a multidimensional model under the form of an OLAP cube [16]. The notation used in the figure was introduced in [17]. In our cube, we define two dimensions: Time and Ontology (Figure 4). The time dimension has two important levels: Week and Day. Metrics by week neutralize the weekly patterns, which contain unsignificant information [18]. More levels can be added depending on the needs (year, months, quarters, ...). The ontology dimension is modeled as a *parent-child dimension* to support ontologies with any number of levels. Other dimensions could be added like physical geography, site geography, Web geography, pages, users, internal referrers, external referrers, or other variations of the time dimension. The cube fact table contains daily term consultation and presence, which are provided by our prototype WASA. The cube measures are consultation, presence, and interest, where the interest measure is a calculated member defined as the division of the first two measures.

4 Experimentation

To test our approach, we developed a prototype called WASA (Figure 5). WASA stands for Web Audience Semantic Analysis. The prototype implements output page collection from Web logs and content journaling (see Section 2) and analyzes

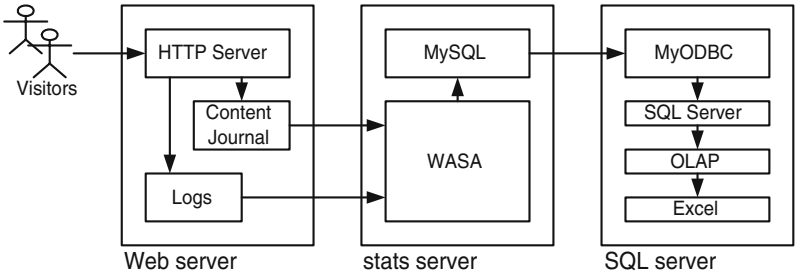

Fig. 5. Experimental configuration

the output and online pages to produce the daily consultation and presence metrics for each term of a given taxonomy. The prototype is written in the Java language and is composed of 10,000 lines of code. The metrics are stored in a MySQL database and transferred into SQL Server for OLAP analysis. In SQL Server OLAP Analysis Service, we introduce the OLAP cube representing the multidimensional model described in Section 3. After cube processing, the metrics are aggregated and can be queried from Microsoft Excel to produce the various visualizations.

4.1 Visualization

In our first case study, we analyzed http://cs.ulb.ac.be, our computer science laboratory's Web site, which contains about 2,000 Web pages and receives an average of 100 page requests a day. The taxonomy was extracted from the ACM classification, which contains 1230 hierarchically-linked terms.

We first produced a multi-line chart where each curve represents the visitors' consultation of the top ACM categories (Figure 6). Computing Methodologies, Software, and Information Systems rank in the top, which is not surprising as these domains are the subject of major lectures. Also, a peak of interest in Theory of Computation can be observed at the beginning of the academic year, when the first-year students start following the corresponding lessons in the computers room. Finally, the average consultation falls down during the academic holiday periods: January-February and July-August. As we can see, this kind of chart can be intuitively related to the problem domain.

To compare the various metrics, we also produced a bar chart representing the metrics for each of the top ACM categories (Figure 7). The most consulted categories are Information Systems, Computing Methodologies, and Software. However, these topics are very present in the Web site, which is confirmed by high presence values. Therefore, high consultation values are not representative of the visitors' interest, for which low interest values can be observed. The interesting topics are Theory of Computation, Data, and Mathematics of Computing. By comparing the consultation and interest in this example, we can see that the considered metrics can dramatically change the ranking of the topics and should be interpreted carefully.

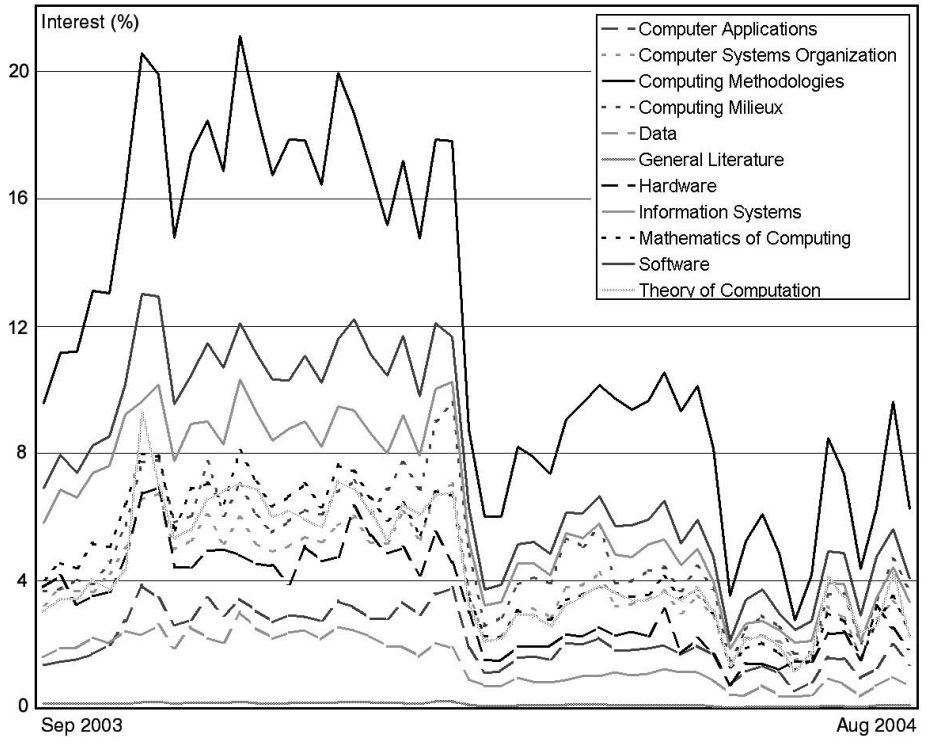

Fig. 6. Consultation of the ACM classification top categories on the cs.ulb.ac.be Web site during the academic year

4.2 Taxonomy Coverage

To test the influence of the taxonomy, we made the same experiments with Eurovoc, the European Commission's thesaurus [19]. Eurovoc contains a taxonomy of 6650 terms, and its domain knowledge include all the European Commission's fields of interest. These include a broad range of domains, from sociology to science, while the ACM classification knowledge domain is focused on computer science. Although Eurovoc contains about five times more terms than the ACM classification, it offers a poor coverage of the computer science domain. Therefore the results obtained with Eurovoc are difficult to relate to the Web site knowledge domain. This kind of problem is typical of very conceptual taxonomies like Eurovoc [19]. This shows how the choice of the taxonomy is important for the results interpretation.

As a natural continuation of the Eurovoc experiment, we studied the benefits of improving taxonomy coverage with respect to the Web site knowledge domain. To evaluate what results can be obtained with an optimal taxonomy enrichment, our department's staff enriched the ACM classification with terms of the Web site. This manual method ensures an optimal improvement of the taxonomy

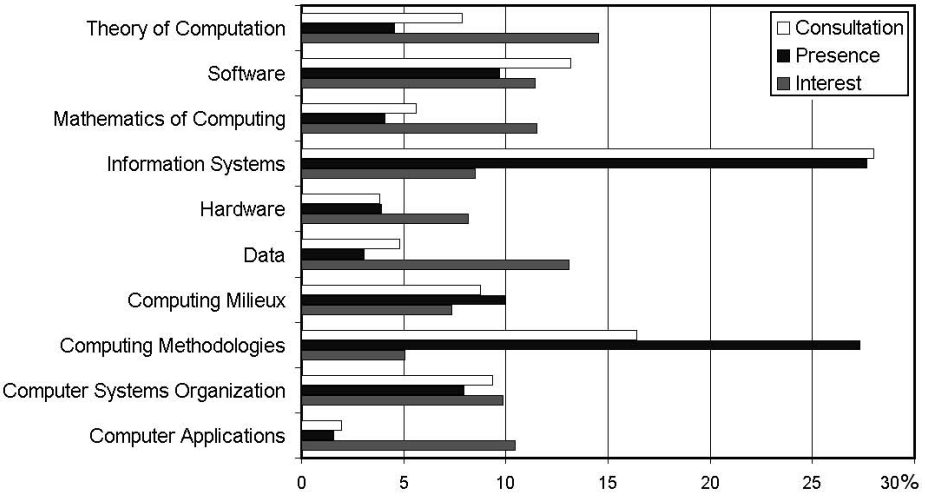

Fig. 7. Audience metrics for the ACM classification top categories

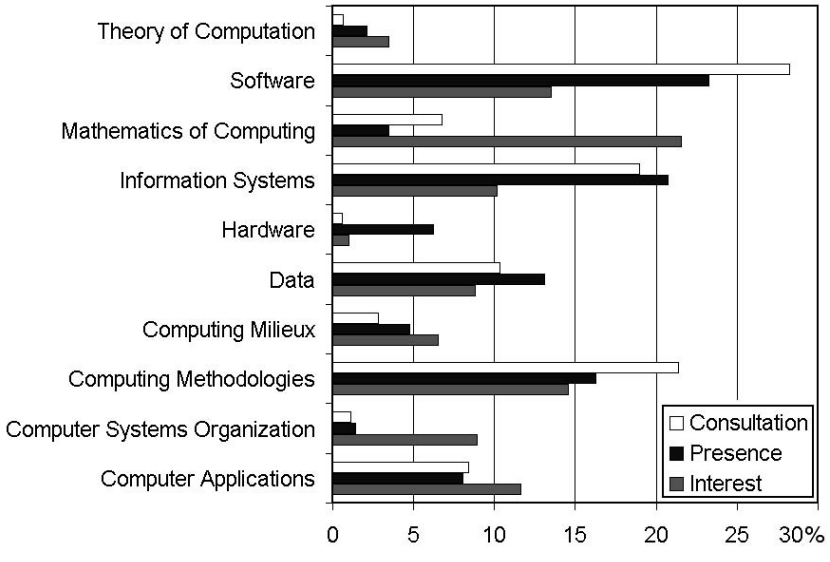

Fig. 8. Audience metrics for the enriched ACM classification top categories

coverage. If we define the taxonomy coverage as the number of taxonomy terms that appear in the output Web pages, our enrichment operations have increased the coverage from 70 to 90, that is an increase of about 30%.

The effect of this enrichment has been evaluated by running WASA with the enriched taxonomy on our department's Web site. With regard to the enriched taxonomy, the top consulted topics are Software, Computing Methodologies, and

Information Systems, while the interesting topics are Mathematics of Computing, Computing Methodologies, and Software (Figure 8). By comparing these results with those obtained with the raw ACM classification (Figure 7), we can see the importance of the taxonomy knowledge domain in the interpretation of the results.

4.3 Validation

To validate our approach against existing software, we compared our results against Google Analytics, a popular Web analytics tool. Although WASA and Google Analytics results are very different, there is a particular case of Web site where the Google Analytics results are comparable to those obtained by WASA. Indeed, if the Web site directories match the taxonomy topics, the hits by directories obtained by Google Analytics should be comparable to the consultation by topic obtained by WASA.

To verify this, we ran the tests on http://wasa.ulb.ac.be, a Web site where the directories have been structured with respect to the Web site topics. For the purpose of the case study, a custom taxonomy containing the main topics and subtopics has been built manually. The Web site main topics include computer science, travel, and leisure. The leisure topic is subdivided into music, chess, cinema, and well-being. The taxonomy contains about 1150 terms in total. The Web site contains about 200 pages and receives about 100 page requests a day.

To compare the results, we produced a directory-based graph with Google Analytics (Figure 9) and a topic-based graph with WASA (Figure 10) representing the audience metrics for the main three topics: computer science, travel, and leisure. By looking at the two graphs, we can see common peaks by the months of March and November. The March peak is due to the referral link from a computer science online magazine, while the November peak is due to the referral link from a music search engine.

The WASA graph in the first trimester of the year shows a predominance of the computer science topic. This predominance cannot be seen in the Google Analytics graph. According to the Web logs, this predominance is due to the success of various computer science pages located outside the computer science directory and linked by computer science site like http://www.linux.org. The dispersion of the pages within the site is rigid because the referral links pointing to these belong to external sites and are not under direct control. The difference in the graphs shows the limitations implied by directory structure rigidity and by page synonymy. Topic-based metrics do not suffer from these limitations and are therefore superior with regard to those aspects.

Another difference can be observed during the November peak, where the travel topic outperforms the leisure topic, while the leisure directory clearly outperforms the travel directory. According to the music content, the success of the travel topic is due to the regions of the world cited in the music pages. This difference in the graphs shows the limitation implied by page polysemy and subsequently the superiority of the topic granularity.

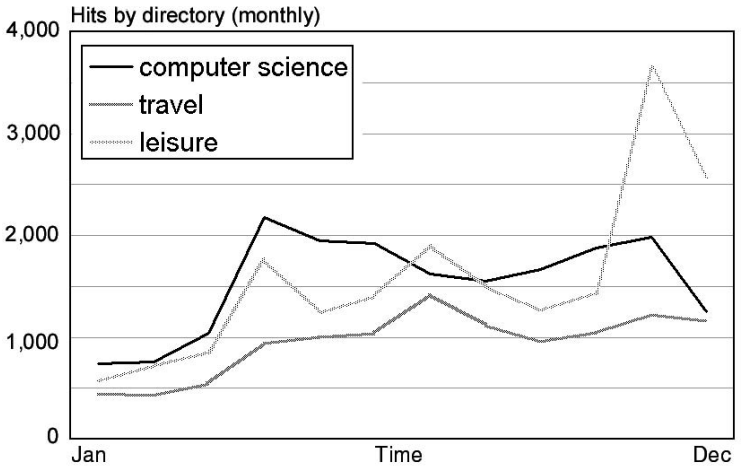

Fig. 9. Directory-based hits obtained with Google Analytics

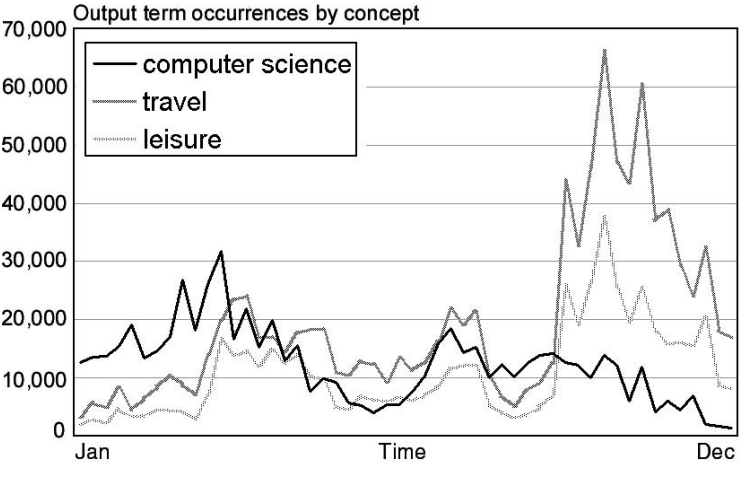

Fig. 10. Topic-based hits obtained with WASA

5 Conclusions and Future Work

In this paper, we presented our solution to answer the need for summarized and conceptual audience metrics in Web analytics. We first described several methods for collecting the Web pages output by Web servers. These methods include content journaling, script parsing, server monitoring, network monitoring, and client-side collection. These techniques can be used alone or in combination to collect the Web pages output by any Web site. Then, we have seen that aggregating the occurrences of taxonomy terms in these pages can provide audience metrics for the Web site topics. According to the first experiments on real data with our proto-

type and SQL Server OLAP Analysis Service, topic-based metrics prove extremely summarized and much more intuitive than page-based metrics.

As a consequence, topic-based metrics can be exploited at higher levels in the organization. For example, organization managers can redefine the organization strategy according to the visitors' interests. Topic-based metrics also give an intuitive view of the messages delivered through the Web site and allow to adapt the Web site communication to the organization objectives. The Web site chief editor on his part can interpret the metrics to redefine the publishing orders and redefine the sub-editors' writing tasks. As decisions at higher levels in the organization should be more effective, topic-based metrics should significantly contribute to Web analytics.

Experiments on real Web sites with several taxonomies like Eurovoc and the ACM classification have shown the importance of the considered metric (consultation, presence, interest) and of the taxonomy coverage of the Web site knowledge domain. Also, comparing our prototype results with a popular Web analytics tool validates our approach while demonstrating the superiority of topic-based metrics over directory-based and page-based metrics. Indeed, these metrics suffer from directory structure rigidity, page synonymy, and page polysemy. This calls for the adoption of topic-based metrics in Web analytics tools.

A limitation to the wide adoption of topic-based metrics is the lack of custom taxonomies for Web sites. To overcome this limitation, we will explore automatic and semi-automatic taxonomy enrichment techniques [20]. In our future work, we will also apply further text analysis techniques to the Web site pages. These techniques will include geo-coding, clustering, date recognition, and organisation/person name identification [21]. The overall analysis will provide a multifacetted vector representation which we will integrate in our multidimensional model. We will also add other dimensions like Web topology and Web site structure. We will evaluate the influence of these additional dimensions by running similar experiments as in this paper. Finally, variations of the metrics inspired from the vector model [14], as well as evaluators for taxonomy coverage of Web site knowledge domain [22], should be experimented to evaluate the taxonomy influence on the results quality.

References

[1] Srivastava, J., Cooley, R., Deshpande, M., Pang-Ning, T.: Web usage mining: Discovery and applications of usage patterns from web data, SIGKDD Explorations 1(2)

[2] March, J., Simon, H., Guetzkow, H.: Organizations, 2nd edn. Blackwell, Cambridge (1983)

[3] Wahli, U., Norguet, J., Andersen, J., Hargrove, N., Meser, M.: Websphere Version 5 Application Development Handbook. IBM Press (2003), `http://www.redbooks.ibm.com/redpieces/pdfs/sg246993.pdf`

[4] Chen, M.-S., Han, J., Yu, P.S.: Data mining: An overview from a database perspective. IEEE Trans. Knowl. Data Eng. 8(6), 866–883 (1996)

[5] Mobasher, B., Cooley, R., Srivastava, J.: Automatic personalization based on Web usage mining. Communications of the ACM 43(8), 142–151 (2000)

[6] Aggarwal, C.C., Yu, P.S.: On disk caching of web objects in proxy servers. In: Proc. of the 6th Int. Conf. on Information and Knowledge Management, CIKM, pp. 238–245 (1997)
[7] Perkowitz, M., Etzioni, O.: Towards adaptive web sites: Conceptual framework and case study. J. of Artif. Intell. 118(1-2), 245–275 (2000)
[8] Büchner, A.G., Mulvenna, M.D.: Discovering internet marketing intelligence through online analytical web usage mining. SIGMOD Record 27(4), 54–61 (1998)
[9] Pirolli, P., Pitkow, J.E.: Distributions of surfers' paths through the world wide web: Empirical characterizations. J. of the World Wide Web 2(1-2), 29–45 (1999)
[10] Ríos, S.A., Velásquez, J.D., Vera, E.S., Yasuda, H., Aoki, T.: Using SOFM to improve web site text content. In: Proc. of the 1st Int. Conf. on Advances in Natural Computation, ICNC, Part II, pp. 622–626 (2005)
[11] Chi, E.H., Pirolli, P., Chen, K., Pitkow, J.E.: Using information scent to model user information needs and actions and the web. In: Proc. of the SIGCHI on Human Factors in Computing Systems, pp. 490–497 (2001)
[12] Facca, F.M., Lanzi, P.L.: Mining interesting knowledge from weblogs: a survey. Data Knowl. Eng. 53(3), 225–241 (2005)
[13] Materna, G.: Extraction par déformattage du contenu de pages Web dynamiques semi-structurées, travail de fin d'études d'Ingénieur civil informaticien, Faculté des Sciences Appliquées, Université Libre de Bruxelles (2002)
[14] Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison-Wesley, Reading (1999)
[15] Stumme, G., Maedche, A.: FCA-MERGE: Bottom-up merging of ontologies. In: Proc. of the 17th Int. Joint Conf. on Artificial Intelligence, IJCAI, pp. 225–234 (2001)
[16] Sweiger, M., Madsen, M., Langston, J., Lombard, H.: Clickstream Data Warehousing. John Wiley & Sons, Chichester (2002)
[17] Malinowski, E., Zimányi, E.: OLAP hierarchies: A conceptual perspective. In: Persson, A., Stirna, J. (eds.) CAiSE 2004. LNCS, vol. 3084, pp. 477–491. Springer, Heidelberg (2004)
[18] Norguet, J.P., Zimányi, E., Steinberger, R.: Improving web sites with web usage mining, web content mining, and semantic analysis. In: Wiedermann, J., Tel, G., Pokorný, J., Bieliková, M., Štuller, J. (eds.) SOFSEM 2006. LNCS, vol. 3831, pp. 430–439. Springer, Heidelberg (2006)
[19] Steinberger, R., Pouliquen, B., Ignat, C.: Exploiting multilingual nomenclatures and language-independent text features as an interlingua for cross-lingual text analysis applications. In: Proc. B of the 7th Int. Multiconference on Language Technologies, IS 2004 (2004)
[20] Maedche, A., Staab, S.: Ontology learning for the semantic web. IEEE Intelligent Systems 16(2), 72–79 (2001)
[21] Steinberger, R., Pouliquen, B., Ignat, C.: Navigating multilingual news collection using automatically extracted information. In: Proc. of the 27th Int. Conf. on Information Technology Interfaces, ITI (2005)
[22] Lozano-Tello, A., Gómez-Pérez, A.: ONTOMETRIC: A method to choose the appropriate ontology. J. of Database Manag. 15(2), 1–18 (2004)